TOWARD ADVANCED TELEOPERATION IN SPACE

Antal K. Bejczy
Senior Research Scientist
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109

INTRODUCTION

The term "teleoperation" refers to the use of manipulators (mechanical arms) and mobility devices, equipped with some sensing capabilities, and remotely controlled by a human operator. Historically, mechanical arms are the most important teleoperated devices.

In general, teleoperation implies <u>continuous</u> perceptive and cognitive human operator involvement in the control of remote manipulators. Typically, the human control is a <u>manual</u> one, and the basic information feedback is through <u>visual images.</u> Continuous human operator involvement in teleoperation has both advantages and disadvantages. The disadvantages become quite dramatic when there is an observable, two-way communication time delay between the operator and the remotely controlled equipment. Modern development trends in teleoperator control technology are aimed at amplifying the advantages and alleviating the disadvantages of the human element in teleoperation through the development and use of various non-visual sensing capabilities, intelligent or task-level computer controls, computer graphics or virtual reality displays, and new computer-based human-machine interface devices and techniques in the information and control channels between the operator and the remotely controlled manipulators. These development trends are typically summarized under the popular titles of telepresence and supervisory control technologies. In this chapter, those two titles are lumped under the term advanced teleoperation.

Several notes should be added to the objective description of telepresence anti supervisory control technologies. First, none of them eliminates the human operator from the control operation, but both change the operator's function assignments and employ human capabilities in new ways. Second, both technologies promise the performance of more complex tasks with better results. But, in doing so, both technologies also make a close reference to human capabilities of operators who will use evolving new devices and techniques in the control station. Third, both telepresence and supervisory control make reference to evolving capabilities of other technologies like sensing, high performance computer graphics, computerized electro-mechanical devices, algorithm-based flexible automation, expert systems for planning and error recovery, and so on. Thus, the progress in both technologies are tied to rich multidisciplinary activities. Fourth, both technologies need the evaluation and validation of their results relative to application environments,

This chapter is focused at the description and some practical evaluation of an experimental advanced teleoperation system, developed at the Jet Propulsion Laboratory (JPL) during the past seven to eight years. First we describe the JPL Advanced Teleoperator (ATOP) system and its control station where a variety of operator interface devices and techniques are integrated into a functional setting, accommodating a primary operator and secondary operators. Then, we will summarize the results of some generic and application task experiments. In the third part of the chapter, we will highlight the lessons learned so far. The chapter will conclude with a brief description of an ongoing work on an anthropomorphic (human-like) advanced telemanipulation system.

JPL ATOP SYSTEM

The basic underlying idea of the JPL ATOP system setting is to provide a dual arm robot system together with the necessary operator interfaces in order to extend the two-handed manipulation

capabilities of a human operator to remote places. The system setting intends to include all perceptive components that are necessary to perform sensitive remote manipulation efficiently, including non-repetitive and unexpected tasks. The general goal is to elevate teleoperation to a new level of task performance capabilities through enhanced visual and non-visual sensing, computer-aided remote control, and computer-aided human-machine interface devices and techniques. The overall system is divided into two major parts: the remote (robot) work site and the local (control station) site, with electronic data and TV communication links between the two sites.

The remote site is a workcell. It comprises: (i) two redundant 8-d.o.f. AAI arms in a fixed base setting, each covering a hemispheric work volume, and each equipped with the latest JPL-developed model C smart hands which contain 3D force-moment sensors at the hands' base and grasp force sensing at the base of the hand claws, (ii) a JPL-developed control electronics and distributed computing system for the two arms and smart hands, and (iii) a computer controllable multi-TV gantry robot system with controllable illumination. This gantry robot currently accommodates three color TV cameras, one on the ceiling plane, one on the rear plane, and one on the right side plane of the workcell. Each camera can be position controlled in two translational d.o.f. in the respective plane, and in two orientation directions (pan and tilt) relative to the respective moving base. Zoom, focus and iris of each TV camera can also be computer controlled. A stereo TV camera system is also available which can be mounted on any of the two side camera bases. The total size of the rectangular remote work site is: about 5 m width, about 4 m depth, and about 2.5 m in height. See Figure 1 for the ATOP remote workcell.

The control station site organization follows the idea of accommodating the human operator in all levels of human-machine interaction, and in all forms of human-machine interfaces. Presently, it comprises: (i) two general purpose Force-Reflecting Hand Controllers (FRHC), (ii) three TV monitors, (iii) TV camera/monitor switchboards, (iv) manual input device for TV control, and (v) three graphics displays: one is comected to the primary graphics workstation (IRIS 4D/310 VGX)

which is used for preview/predictive displays and for various graphical user interfaces (GUI'S) in four-quadrant format; the second is connected to an IRIS 4D/70 GT workstation and is solely used for sensor data display; the third one is connected to a SUN workstation (SparcStation 10) and is used as a control configuration editor (CCE), which is an operator interface to the manipulators' control software based on X-window environment. See Figure 2 for the ATOP local control station.

ATOP Hand Controllers

The human arm and hand are functionally both powerful mechanical tools and delicate sensory organs through which information is received from and transmitted to the world. Therefore, the human arm-hand system (thereafter simply called hand here) is a key communication medium in teleoperator control. With hand actions, complex position, rate or force commands can be formulated and very physically written to the controller of a remote robot arm system in all workspace directions. At the same time, the human hand also can receive force, torque, and touch information from the remote robot arm-hand system. Furthermore, the human fingers offer additional capabilities to convey new commands to a remote robot controller from a suitable hand controller. Hand controller technology is, therefore, an important technology in the development of advanced teleoperation. Its importance is particularly underlined when one considers computer control which connects the hand controller to the remote arm system, The direct and continuous (scaled or unscaled) relation of operator hand motion to the remote robot arm's motion behavior in real time through a hand controller is in sharp contrast to the computer keyboard type commands which, by their very nature, are symbolic, abstract and discrete (non-continuous), and require the specification of some set of parameters within the context of a desired motion,

4

In contrast to the standard force-reflecting, replica master-slave systems, a new form of bilateral, force reflecting manual control of robot arms has been implemented at the JPL ATOP project. The hand controller is a backdrivable six d.o.f. isotonic joystick. It is dissimilar to the controlled robot arm both kinematically and dynamically. But, through computer transformations, it can control' (motion of any robot arm in six task space coordinates (in three position and three orientation coordinates). Forces and moments sensed at the base of the robot hand can back-drive the hand controller through proper computer transformations so that the operator feels the forces and moments acting at the robot hand while he controls the position and orientation of it. This hand controller can read the position and orientation of the hand grip within a 30 cm cube in all orientations, and can apply arbitrary force and moment vectors up to 20 N and 1.0 Nm, respectively, at the hand grip.

The overall schematic of the six-degree-of-freedom Force-Reflecting Hand Controller (FRHC) is shown in Fig. 3. (The mechanism of the hand controller was designed by J. K. Salisbury, Jr., now at MIT, Cambridge, MA). The kinematics and the command axes of the FRHC are shown in Fig. 4. The hand grip is supported by a gimbal with three intersecting axes of rotation (β_4, f35, β_6). A translation axis (R_3) connects the hand gimbal to the shoulder gimbal which has two more intersecting axes (β_1, β_2). The motors for the three hand gimbal and translation axes are mounted on a stationary drive unit at the end of the hand controller's main tube. This stationary drive unit forms a part of the shoulder gimbal's counterbalance system. The moving part of the counterbalance system is connected to the R_3. It serves (i) to maintain the hand controller's center of gravity at a fixed point and (ii) to maintain the tension in the hand gimbal's drive cables as the hand gimbal changes its distance from the stationary drive unit. The actuator motors for the two shoulder joints are mounted to the shoulder gimbal frame and to the base frame of the hand controller, respectively.

The self-balance system renders the hand controller neutral against gravity. Thus, the hand controller can be mounted both horizontally or vertically, and the calculation of motor torques to back-drive the hand controller does not require gravity compensation. In general, the mechanical design of the hand controller provides a dynamically "transparent" input/output device for the operator. This is accomplished by low backlash, low friction and low effective inertia at the hand grip. More details of the mechanical design of this hand controller and on hand controller technology in general can be found in [1-2]. A computer-based control system establishes the appropriate kinematic and dynamic control relations between the FRHC and the robot arm. The FRHC can control any robot arm and can receive force/torque feedback from any robot arm equipped with 3D force-moment sensor at the base of the robot hand.

The computer-based control system supports four modes of manual control: position, rate, force-reflecting, and compliant control in task space (Cartesian space) coordinates. The operator, through an on-screen menu, can designate the control mode for each task space axis independently. Position control mode servos the slave position and orientation to match the master's. The indexing function allows slave excursions larger or smaller than the 30 cm cube hand controller work volume. In force-reflecting mode, the hand controller is back-driven based on force-moment data generated by the robot and sensed during the robot hand's interaction with objects and environment. Rate control mode sets slave endpoint velocity in task space based on the displacement of the hand controller. This is implemented through a software spring in the control computer of the hand controller. Through this software spring, the operator has a sensation of the commanded rate, and the software spring also provides a zero-referenced restoring force. Rate mode is useful for tasks requiring large translations. Compliant control mode is implemented through a low-pass software filter acting on the robot hand's force-torque sensor data in the hybrid position-force loop. This permits the operator to control a springy or less stiff robot. Active compliance with damping can be varied by changing the filter parameters in the software menu. Setting the spring parameter to zero in the low pass filter will reduce it to

6

a pure damper which results in a high stiffness hybrid position-force control loop.

The present FRHC has a simple hand grip equipped with a deadman switch and with three function switches. In order to better utilize the operator's finger input capabilities, an exploratory project recently evaluated a design concept that would place computer keyboard features attached to the hand grip of the FRHC. To accomplish this, three DATA HAND™[3] switch modules were integrated with the hand grip as shown in Fig. 5. Each switch module at a finger tip contains five switches as indicated in Fig. 6. Thus, the three switch modules at the FRHC hand grip can contain fifteen function keys which directly can communicate with a computer terminal. This eliminates the need that the operator moves his/her hand from the FRHC hand grip to a separate keyboard to input messages and commands to the computer. A recent test and evaluation, using a mock-up system and ten test subjects, indicated the viability of the finger-tip switch modules as part of a new hand grip unit for the FRHC as a practical step towards a more integrated operator interface device in the ATOP system. More on this concept and evaluation can be found in [4].

ATOP Control System

The overall ATOP control organization permits a spectrum of operations between full manual, shared manual and automatic, and full automatic (call traded) control, and the control can be operated with variable active compliance referenced to force-moment sensor data. More on the ' overall ATOP control system can be found in [5-8]. Only the salient features of the original ATOP control system are summarized here. The overall control /information data flow diagram (for a single arm) is shown in Fig. 7. It is noted that the computing architecture of this original ATOP system is a fully synchronized pipeline, where the local servo loops at both the control station and the remote manipulator nodes can operate at 1000 Hz rate. The end-to-end bilateral (i.e., force-reflecting) control loop can operate at 200 Hz rate, More on the computational system

7

critical path functions and performance can be found in [9].

The actual data flow depends on the control mode chosen. The different selectable control modes are the following: freeze mode; neutral mode; current mode; joint mode; task mode. In Freeze mode the brakes of joints are locked, the motors are turned off, and some joints are servoed to maintain their last positions. This mode is primarily used when the robot is not needed for a short period of time but turning it off is not desired. In Neutral mode all position gains are set to zero, gravity compensation is active to prevent the robot from falling down. In this mode the user can manually move the robot to any position and it will stay there. In Current mode the six motor currents are directly commanded by the data coming in from the communication link. This mode exists for debugging only. In Joint mode the hand controller axes control individual motors of the robot. In, Task mode the inverse kinematic transformation is performed on the incoming data, and the hand controller controls the end effecter tip along the three Cartesian and pitch, yaw and roll axes. This mode is the most frequently used for task execution or experiments, and this is the one shown explicitly in Fig. 7.

The control system on the remote site is designed to prevent sudden robot motions. The motion commands received are incremental and are added to the current parameter under control. Sudden large motions are also prevented in case of mode changes. This necessitates proper . initialization of the inverse kinematics software at the time of the mode transition. This is done by inputing the current Cartesian coordinates from the forward kinematics into the inverse kinematics.

The data flow diagram shown in Fig. 7 illustrates the organization of several servo loops in the system. The imermost loop is the position control servo at the robot site. This servo uses a PD control algorithm, where the damping is purely a function of the robot joint velocities. The incoming data to this servo is the desired robot trajectory described as a sequence of points at 1

8

msec intervals. This joint servo is augmeneted by gravity compensation routine to prevent the weight of the robot from causing joint positioning error. Since this servo is a first order one, there will be a constant position error that is proportional to the joint velocity.

In basic Cartesian control mode the data from the hand controller are added to the previous desired Cartesian position. From this the inverse kinematics generates the desired joint positions. The joint servo moves the robot to this position. From the actual joint position the forward kinematics computes the actual Cartesian positions. The force torque sensor data and the actual positions are fed back to the hand controller side to provide force feedback.

This basic mode can be augmented by the addition of Compliance control, Cartesian servo, and Sticktion/Friction compensation. Figure 8 shows the Compliance control and the Cartesian servo augmentations. There are two forms of compliance, integrating and spring type (see Fig. 9). In integrating compliance the velocity of the robot end effecter is proportional to the force felt in the corresponding direction. To eliminate drift a dead-band is used. The zero velocity band does not have to be a zero force, a force offset maybe used. Such a force offset is used if, for example, when we want to push against the task board at some given force while moving along other axes. Any form of compliance can be selected along any axis independently. In case of the spring type compliance the robot position is proportional to the sensed force. This is similar to a spring centering action. The velocity of the robot motion is limited in both the integrating and spring cases.

There is a wide discrepancy between the robot response bandwidth and the force readings. The forces are read at a 1000 Hz sampling rate. The robot motion command has an output response at a 5 Hz bandwidth. To generate smooth compliance response, the force readings go through two subsequent filters. The first one is a simple averaging of ten force readings. This average is called 100 Hz force and is computed at a 100Hz rate. From this 100 Hz force a 5 Hz force reading is

9

computed by a first order low pass filter. This 5 Hz force reading is also computed at a 100 Hz rate. The 5 Hz force is used for compliance computations.

As shown in Fig. 8, the Cartesian servo acts on task space (X, Y, Z, pitch, yaw, roll) errors directly. These errors are the difference between desired and actual task space values. The actual task space values are computed from the forward kinematic transformation of the actual joint positions. This error is then added to the new desired task space values before the inverse kinematic transformation determines the new joint position commands from the new task space commands.

A trajectory generator algorithm was formulated based on observations of profiles of task space trajectories generated by the operators manually through the FRHC. Three important features were observed in hand-generated task space trajectory profiles: (1) The operators always generated trajectories as a function of the relative distance between start point and goal point in the task space or, in general, as a function of the present position state relative to the desired position state of the end effecter in the task space. In other words, the operators manually do not generate trajectories based on time (on clock signals). (2) The velocity-position phase diagrams of motion typically resembled a harmonic (sine) function. (3) Between the start and completion phases, the operator-generated trajectories typically attained a constant velocity profile.

Based on these observations, we formulated a Harmonic Motion Generator (HMG) with a sinusoidal velocity-position phase function profile as shown in Fig. 10. The motion is parameterized by the total distance traveled, the maximum velocity, and the distance used for acceleration and deceleration. Both the accelerating and decelerating segments are quarter sine waves, with a constant velocity segment connecting them, This scheme still has a problem, the velocity being O before the motion starts. This problem is corrected by adding a small constant to the velocity function,

10

It is noted that the HMG discussed here is quite different from the typical trajectory generator algorithms employed in robotics which use a polynomial position-time function. Our algorithm generates the motion as a trigonometric (harmonic) velocity *versus* position function. The position versus time and the corresponding velocity versus time functions generated by the HMC; are shown in Fig. 11. More on performance results generated by HMG, Cartesian servo and force-torque sensor data filtering in compliance control can be found in [6] and [10]. Illustrative examples are shown in Figures 12 and 13.

ATOP Computer Graphics

Task visualization is a key problem in teleoperation, since most of the operator's control decisions are based on visual or visually conveyed information. For this reason, computer graphics plays an increasingly important role in advanced teleoperation. This role includes: (i) planning actions, (ii) previewing motions, (iii) predicting motions in real time under communication time delay, (iv) to helping operator training, (v), to enabling visual perception of non-visible events like forces and moments, and (vi) to serving as a flexible operator interface to the computerized control system.

The capability of task planning aided by computer graphics offers flexibility, visual quality and a quantitative design base to the planning process. The capability of graphically previewing motions enhances the quality of teleoperation by reducing trial-and-error strategies in the hardware control and by increasing the operator's confidence in control decision making during task execution. Predicting consequences of motion commands in real time under communication time delay permits longer action segmentations as opposed to the move-and-wait control strategy normally employed when no predictive display is available, increases operation safety, and reduces total operation time. Operator training through a computer graphics display system is a convenient tool for familiarizing the operator with the teleoperated system without turning the

11

hardware system on. Visualization of non-visible effects (like contact forces) enables visual perception of different non-visual sensor data, and helps management of system redundancy by providing some suitable geometric image of a multi-dimensional system state. Last, but not least, computer graphics as a flexible operator interface to the control systems replaces complex switchboard and analog display hardware in a control station.

The actual utility of computer graphics in teleoperation to a high degree depends on the fidelity of graphics models that represent the teleoperated system, the task and the task environment. The JPL ATOP project in the past few years developed high-fidelity calibration of graphics images to actual TV images of task scenes. This development has four major ingredients. First, creation of high-fidelity 3-D graphics models of robot arms and objects of interest for robot arm tasks. Second, high-fidelity calibration of the 3-D graphics models relative to given TV camera 2-D image frames which cover the sight of both the robot arm and the objects of interest. Third, high-fidelity overlay of the calibrated graphics models over the actual robot arm and object images in a given TV camera image frame on a monitor screen. Fourth, high-fidelity motion control of robot arm graphics image by using the same control software that drives the real robot.

The high fidelity fused virtual and actual reality image displays became very useful tools for planning, previewing and predicting robot arm motions without commanding and moving the robot hardware. The operator can generate visual effects of robot motion by commanding and controlling the motion of the robot's graphics image superimposed over TV pictures of the live scene. Thus, the operator can see the consequences of motion commands in real time, before sending the commands to the remotely located robot. The calibrated virtual reality display system can also provide high-fidelity synthetic or artificial TV camera views to the operator. These synthetic views can make critical motion events visible that otherwise are hidden from the operator in a given TV camera view or for which no TV camera view is available. More on the graphics system in the ATOP control station can be found in[11] through [15].

12

A high-fidelity overlay of graphics and TV images of work scenes requires a high fidelity TV camera calibration and object localization relative to the displayed TV camera view. Theoretically, this can be accomplished in several ways. For the purpose of simplicity and operator-controllable reliability, an operator-interactive camera calibration and object localization technique has been developed, using the robot arm itself as a calibration fixture, and using a non-linear least-squares algorithm combined with a linear one as a new approach to compute accurate calibration and localization parameters.

The current method uses a point-to-point mapping procedure, and the compilation of camera parameters is based on the ideal pinhole model of image formation by the camera. In the camera calibration procedure, the operator first enters the correspondence information between the 3-D graphics model points and the 2-D camera image points of the robot arm to the computer. This is performed by repeatedly clicking with a mouse a graphics model point and its corresponding TV image point for each corresponding pair of points on a monitor screen which, in a four-quadrant window arrangement, shows both the graphics model and the actual TV camera image. (See Figure 14). To improve calibration accuracy, several poses of the manipulator within the same TV camera view can be used to enter corresponding graphics model and TV image points to the computer. Then the computer computes the camera calibration parameters. Because of the ideal pinhole model assumption, the computed output is a single linear 4 by 3 calibration matrix for a linear perspective projection.

Object localization is performed after camera calibration by entering corresponding object model and TV image points to the computer for different TV camera views of the object. Again, the computational output is a single linear 4 by 3 calibration matrix for a linear perspective projection.

13

The actual camera calibration and object localization computations are carried out by a combination of linear and non-linear least-squares algorithms. The linear algorithm, in general, does not guarantee the orthonormality of the rotation matrix, providing only an approximate solution. The non-linear algorithm provides the least-squares solution that satisfies the orthonormality of the rotation matrix, but requires a good initial guess for a convergent solution without entering into a very time-consuming random search. When a reasonable approximate solution is known, one can start with the non-linear algorithm directly. When an approximate solution is not known, the linear algorithm can be used to find one, and then one can proceed with the non-linear algorithm. More on the calibration and object localization technique can be found in [16, 17].

After completion of camera calibration and object localization, the graphics models of both robot arm and object of interest can be overlaid with high fidelity on the corresponding actual images of a given TV camera view. The overlays can be in wire-frame or solid-shaded polygonal rendering with varying levels of transparency, providing different task details. In the wire-frame format, the hidden lines can be removed or retained by the operator, dependent on the information needs in a given task.

ATOP Graphics Operator Interface

The first development of a graphic system as an advanced operator interface was aimed at parameter acquisition, and was handled and called as a Teleoperation Configuration Editor (TCE) [18], This interface used the concepts of Windows, Icons, Menus, and Pointing Device to allow the operator to interact, select, and update single parameters as well as groups of parameters. TCE utilizes the direct manipulation concept, with the central idea to have visible objects such as buttons, sliders, and icons that can be manipulated directly, i.e. moved and selected using the mouse, to perform any operation. A graphic interface of this type has several

14

advantages over a traditional panel of physical buttons, switches, and knobs: the layout can be easily modified and its implementation cycle, i.e., design and validation, is significantly shorter than hardware changes.

The TCE, Fig. 15, was developed to incorporate all the configuration parameters of an early single arm version of the ATOP system. It was organized in a single menu divided in several areas dedicated to the parameters of a specific function. Dependencies among different graphical objects are embedded in the interface so that, when an object is activated, the TCE checks for parameters congruency. A significant feature of this implementation is the capability of storing and retrieving sets of parameters via macro buttons. When a macro command is invoked, it saves the current system configuration and stores it in a function button which can later restore it. The peg-in-hole task, for instance, requires mostly translational motions but when holes have a tight clearance, a compliance is necessary. An appropriate macro configuration is one that enables x, y, and z axes, with position control in the approach direction and automatic compliance on the other two axes. This configuration can be assigned to a macro button and then recalled during a task containing a peg-in-hole segment.

The continuing work on a graphic system as an advanced operator interface is aimed at the data presentation structure of the interface problem, and, for that purpose, uses a hierarchical architecture [15]. This hierarchical data interface looks like a menu tree with only the last menu of the chain (the leaf) displaying data. All the ancestors of the leaf are visible to clearly indicate the nature of the data displayed. The content of the leaf includes data or pictures and quickly conveys the various choices available to the operator. A schematic figure of this layout is shown in Figure 16. Parameters have been organized in four large groups that follow the sequence of steps in a teleoperation protocol. These groups are: (I) Layout, (ii) #configuration, (iii) Tools, (iv) u " Execution. Each group is further subdivided into specific functions. The Layout menu tree contains the parameters defining the physical task structure, such as relative position of the

robots and of the FRHC, servo rates, etc. The Configuration menu tree contains the parameters

necessary to define task phases, such as control mode and control gains. The Tools tree contains

parameters and commands for the off-line support to the operator, such as planning, redundancy

resolution and software development. Finally, the Execution tree contains commands and

parameters necessary while teleoperating the manipulators, such as data acquisition, monitoring

of robots, hand controllers and smart hands, retrieval of stored configurations and camera

commands.

This hierarchical data interface helps solving the problem of displaying the large amount of data

needed during a teleoperation task, but it does not address the issue of data entry by the

operator.' Traditional interfaces require serveral operators, using a different input device for each

controlled function. A single operator using this interface needs to use different input devices:

joysticks to move the manipulators, buttons for camera and video control, keyboard and mouse

for parameters entry. Operating these devices is time consuming and distracts the operator from

the task.

An alternative approach would be using the FRHC as the only input device for the operator. This

scheme is similar to a virtual reality interface, where operators can accesss commands and data

by simply moving their arms and hands. In our implementation concept, the operator would use

the FRHC as a pointing device on the interface menus and would use the FRHC trigger to click

the selected button. The discrimination between commands for the robot and those for the data

interface could be done by the deadman switch on the FRHC handle. When not pressed, this

switch inhibits the generation of motion commands for the robots, but FRHC motion data are still

available and can be used to move the cursor. This scheme also can be combined with the

DATAHAND™ switch module integrated with the FRHC hand grip, discussed previously in this

chapter.

16

ATOP CONTROL EXPERIMENTS

To evaluate advanced teleoperation capabilities, two types of experiments were designed and conducted: experiments with generic tasks and experiments with application tasks. Generic tasks are idealized, simplified tasks and serve the purpose of evaluating some specific advanced teleoperation features. Application tasks are simulating some real-world use of advanced teleoperation.

Generic Task Experiments

In these experiments, described in detail in [19], four tasks were used: attach and detach velcro; peg insertion and extraction; manipulating three electrical connector; manipulating a bayonet connector. Each task was broken down to subtasks. The test operators were chosen from a population with some technical background but not with an in-depth knowledge of robotics and teleoperation. Each test subject received two to four hours of training on the control station equipment. The practice of individuals consisted of four to eight 30-minute sessions.

As pointed out in [19], performance variation among the nine subjects was surprisingly slight. Their backgrounds were similar (engineering students or recent graduates) except for one who was a physical education major with training in gymnastics and coaching. This subject showed the best overall performance by each of the measures. This apparent correlation between performance and prior background might suggest that potential operators be grouped into classes based on interest and aptitudes.

The generic task experiments were focused at the evaluation of kinesthetic force feedback versus no force feedback, using the specific force feedback implementation techniques of the JPL ATOP project. A typical generic experiment is shown in Figure 17. The evaluation of the experimental

data supports the idea that multiple measures of performance must be used to characterize human performance in sensing and computer aided teleoperation. For instance, in most cases kinesthetic force feedback significantly reduced task completion time. In some specific cases, however, it did knot, but it did sharply reduce extraneous forces. More on the results in [19-20], *see also chapter in this book on "Ground Experiments Towards Space Teleoperation with Time Delay" by B. Hannaford.*

Application Task Experiments

Two major experiments were performed: one without communication time delay and one with communication time delay.

The experiments without communication time delay were grouped around a simulated satellite repair task. The particular repair task was the duplication of the Solar Maximum Satellite Repair (SMSR) mission, which was performed by two astronauts in Earth orbit in the Space Shuttle Bay in 1984. Thus, it offers a realistic performance reference data base. This repair is a very challenging task, since this satellite was not designed for repair. Very specific auxiliary substasks must be performed (e.g. a hinge attachment) in order to accomplish the basic repair which, in our simulation, is the replacement of the Main Electric Box (MEB) of the satellite. The total repair, as performed by two astronauts in Earth orbit, lasted for about three hours, and comprised the following set of substasks: thermal blanket removal, hinge attachment for MEB opening, opening of the MEB, removal of electrical connectors, replacement of MEB, securing parts and cables, replug of electrical connectors, closing of MEB, reinstating thermal blanket. It is noted that the two astronauts were trained for this repair on the ground for about a year.

The SMSR simulation by ATOP capabilities was organized so that each repair scenario had its own technical justification and performance evaluation objective. For instance, in the first subtask-scenario performance experiments, alternative control modes, alternative visual settings, operator skills versus training, and evaluation measures themselves were evaluated [21, 22]. The

first subtask-scenario performance experiments involved thermal blanket cutting and reinstating, and unscrewing MEB bolts. That is, both subtasks implied the use of tools. Figure 18 illustrates these experiments.

Several important observations were made during the above-mentioned subtask-scenario performance experiments. The two most important ones are: (i) the remote control problem in any teleoperation mode and using any advanced component or technique is at least in 50% a visual perception problem to the operator, influenced greatly by view angle, illumination and contrasts in color or in shading, (ii) the training or, more specifically, the training cycle has a dramatic effect upon operator performance. It was found that the first cycle should be regarded as a familiarization with the system and with the task. For a novice operator, this familiarization cycle should be repeated at least twice. The real training for performance evaluation can only start after completion of a familiarization cycle. The familiarization can be considered as completed when the "trainee understand the system 1/0 details, the system response to commands, and the task sequence details. During the second cycle of training, performance measurements should be made so that the operator understands the content of measures against which the performance will be evaluated. Note, that it is necessary to separate each cycle and repetitions within cycles by at least one day. Once a personal skill has been formed by the operator as a consequence of the second training cycle, the real performance evaluation experiments can start. A useful criteria for determining the sufficient level of training can be, for instance, that of computing the ratio of standard deviation of completion time to mean completion time (that is, computing the coefficient of variation), If the coefficient of variation of the last five trials of a subtask performance is less than 20%, than sufficient level of training can be declared. In the above quoted subtask scenario experiments, the real training, on the average, required one week per subject. More details on application task experiments can be found in [21, 22].

The practical meaning of training is, in essence, to help the operator develop a mental model of the system and of the task. During task execution, the operator acts through the aid of this mental model. It is, therefore, critical that the operator understands very well the response characteristics of the sensing and computer-aided ATOP system which has a variety of selectable control modes, adjustable control gains and scale factors.

The procedure of operator training and the expected behavior of a skilled operator following an activity protocol offers the idea of providing the operator with performance feedback messages on the operator interface graphics, derived from a stored model of the task execution. A key element for such advanced performance feedback tool to the operator is a program that can follow the evolution of a teleoperated task by segmenting the sensory data stream into appropriate phases.

A task segmentation program of this type has been implemented by means of a Neural Network architecture [23] and it is able to identify the segments of a peg-in-hole task. With this architecture, the temporal sequence of sensory data generated by the wrist sensor on the manipulators are turned into spatial patterns and a window of sensor observations which is related to the current task phase. A Partially Recurrent Network algorithm was employed in the computation. Partially Recurrent Networks represent well the temporal evolution of a task, since they include in the input layer a set of nodes connected to the output units, to create a context memory. These units represents the task phase already executed - the previous state. Several experiments of the peg-in-hole task have been carried out and the results have been encouraging with a percentage of correct segmentations approximately equal to 65%. More on these experiments can be found in [23, 24].

The performance experiments with communication time delay conducted on a large laboratory scale in early 1993, utilized a simulated life-size satellite servicing task which was set up at the

20

Goddard Space Flight Center (GSFC) and controlled 4000 km away from the JPL ATOP control station. Three fixed. camera settings were used at the GSFC worksite, and TV images were sent to the JI'L control station over the NASA-Select Satellite TV channel at video rate. Command and control data from JPL to GSFC and status and sensor data from GSFC to JPL were sent through the Internet computer communication network. The roundtrip command/information time delay varied between four to eight seconds between the GSFC worksite and the JPL control station, dependent on the data communication protocol.

The task involved the exchange of a satellite module. This required inserting a 45 cm long power screwdriver, attached to the robot arm, through a 45 cm long hole to reach the module's latching mechanism at the module's backplane, unlatching the module from the satellite, connecting the module rigidly to the robot arm, and removing the module from the satellite. The placement of a new module back to the satellite's frame followed the reverse sequence of actions.

Four camera views were calibrated for this experiment, entering 15 to 20 correspondence points in total from 3 to 4 arm poses for each view. The calibration and object localization errors at the critical tool insertion task amounted to about 0,2 cm each, well within the allowed insertion error tolerance. This 0.2 cm error is referenced to the zoom-in view (fovy=8°) from the overhead (front view) camera which was about 1 m away from the tool tip. For this zoom-in view, the average error on the image plane was typically 1.2 to 1.6% (3.2 to 3.4% maximum error); a 1.4% average error is equivalent to 0.2 cm displacement error on the plane 1 m in front of the camera.

The idea with the high-fidelity graphics image over a real TV image is that the operator can interact with it visually in real time on a monitor within one perceptive frame when generating motion commands manually or by a computer algorithm. Thus, this method compensates in real time for the operator's visual absence from reality due to the time-delayed image. Typically, the geometric dimensions of a monitor and the geometric dimensions of the real work scene shown

21

on the monitor are quite different. For instance, and 8-inch long trajectory on a monitor can correspond to a 24 inch long trajectory in the actual work space, that is, three times longer than the apparent trajectory on the monitor screen. Therefore, to preserve fidelity between previewed graphics arm image and actual arm motions, all previewed actions on the monitor were scaled down very closely to the expected real motion rate of the arm hardware. The manually generated trajectories were also previewed before sending the motion commands to the GSFC control system in order to verify that all motion data were properly recorded. Preview displays contribute to operational safety. In order to eliminate the problem associated with the varying time delay in data transfer, the robot motion trajectory command is not executed at the GSFC control system until all the data blocks for the trajectory are received. An element of fidelity between graphics arm image and actual arm motion was given by the requirement that the motion of the graphics image of the arm on the monitor screen be controlled by the same software that controls the motion of the actual arm hardware. This required to implement the GSFC control software in the JPL graphics computer.

A few seconds after the motion commands were transmitted to GSFC from JPL, the JPL operator could view the motion of the real arm on the same screen where the graphics arm image motion was previewed. If everything went well, the image of the real arm followed the same trajectory on the screen that the previewed graphics arm image motion previously described, and the real arm image motion on the screen stopped at the same position where the graphics arm image motion stopped earlier. After completion of robot arm motion, the graphics images on the screen were updated with the actual final robot joint angle values. This update eliminates accumulation of motion execution errors from the graphics image of robot arm, and retains graphics robot arm position fidelity on the screen even after the completion of a force sensor referenced compliance control action.

The actual contact events (moving the tool within the hole and moving the module out from or in to the satellite's frame) were automatically controlled by an appropriate compliance control algorithm referenced to data from a force-moment sensor at the end of the robot arm, implemented by the cooperating GSFC team and invoked by the JPL operator when needed.

The experiments have been performed successfully, showing the practical utility of high-fidelity predictive-preview display techniques, combined with sensor-referenced automatic compliance control, for a demanding telerobotic servicing task under communication time delay. More on these experiments and on the related error analysis can be found in [16, 17]. Figure 19 illustrates a few typical overlay views.

A few notes are in place here, regarding the use of calibrated graphics overlays for time-delayed remote control. (i) There is a wealth of computation activities that the operator has to exercise. This requires very careful design considerations for an easy and user friendly operator interface to this computation activity. (ii) The selection of the matching graphics and TV image points by the operator has an impact on the calibration resul ts. First, the operator has to select significant points. This requires some rule-based knowledge about what is a significant point in a given view. Second, the operator has to use good visual acuity to click the selected significant points by the mouse.

LESSONS LEARNED

The following general conclusions emerged so far from the development and experimental evaluation of the JPL ATOP:

1. The sensing, computer and graphics aided advanced teleoperation system truly provides
   new and improved technical features. In order to transform these features into new and

23

improved task performance capabilities, the operators of the system have to be transformed from naive to skilled operators. This transformation is primarily an undertaking of education . and training.

2. To carry out an actual task requires that the operator follows a clear procedure or protocol which has to be worked out off-line, tested, modified and finalized. It is this procedure or protocol following habit that finally will help develop the experience and skill of an operator.

3. The final skill of an operator can be tested and graded by the ability of successfully improvising to recover from unexpected errors in order to complete a task.

4. The variety of 1/0 activities in the ATOP control station requires workload distribution between two operators. The primary operator controls the sensing and computer aided robot arm system, while the secondary operator controls the TV camera and monitor system and assures protocol following. Thus, the coordinated training of two cooperating operators is essential to successfully use the ATOP system for performing realistic tasks. It is yet not know what a single operator could do and how. To configure and integrate the current ATOP control station for successful use by a single operator is a challenging R&D work.

5. The problem of ATOP system development is not only to find ways to improve technical components and to create new subsystems, The final challenge is to integrate the improved or new technical features with the natural capabilities of the operator through appropriate human-machine interface devices and techniques to produce an improved overall system performance capability in which the operator is part of the svstem in some new way.

PERSPECTIVES OF ANTHROPOMORHIC TELEMANIPULATION

The robot arms employed in the JPL ATOP project are of industrial type with industrial type parallel claw end effecters. This sets definite limits for the arms' task performance capabilities since dexterity in manipulation resides in the mechanical and sensing capabilities of the hands (or end effecters). The use of industrial type arms and end effecters in space would essentially require to design space manipulation tasks matching the capabilities of industrial type arms and end effecters. Contrary to that, existing space manipulation tasks (except the handling of large space cargos) are designed for astronauts, including the tools to be used by astronauts. There are well over two hundred tools that today are available and certified for use by Extra Vehicular Activity (EVA) astronauts in space. Motivated by these facts, an effort parallel to the ATOP project was initiated at JPL to develop and evaluate human-equivalent or human-rated dexterous telemanipulation capabilities for potential applications in space since all manipulation related tools used by EVA astronauts are human rated.

The general technical approach adopted in this anthropomorphic telemanipulation project is the development and evaluation of an anthropomorphic (human-like) exoskeleton master-slave force-reflecting arm-hand system, This technical approach' implies the following: (i) the master arm is a replica of the slave arm, and each arm has 'seven"degrees of freedom, (ii) the master arm is solidly attached to the operator's arm, (iii) forces acting on the slave arm can backdrive the master arm so that the operator can feel the forces/moments acting on the slave arm, (iv) the slave arm is a human-like fingered hand with a replica glove-like master controller attached solidly to the operator's hand, and (v) forces acting on the slave fingers can backdrive the fingers of the master glove so that the operator can feel the forces acting on the slave fingers. The capability that the operator can feel forces acting at the remote slave site provides kinesthetic telepresence to the operator. This enables that operator to perform sensitive, force-compliant manipulation tasks with or without tools.

The actual design and laboratory prototype development included the following specific technical features: (i) the system is fully electrically driven; (ii) the hand and glove have four fingers (little finger is omitted) and each finger has four degrees of freedom, (iii) the base of the slave fingers follow the curvature of the human finger s hand; (iv) the slave hand and wrist form a mechanically integrated closed subsystem, that is, the hand cannot be used without its wrist; (v) the lower slave arm which connects to the wrist houses the full electromechanical drive system for the hand and wrist (altogether 19 degrees of freedom), including control electronics and microprocessors, and (vi) the slave drive system electromechanically emulates the dual function of human muscles: position and force control. This implies a novel and unique implementation of active compliance. Taken all the specific technical features together makes this exoskeleton unique among the few similar systems. No previous or ongoing other developments have all the above quoted technical features in one integrated system, and some of the specific technical features are not represented in any other similar system at all. More on this system can be found in [25].

Currently, the JPL anthropomorphic telemanipulation system is assembled and tested in a "terminus control configuration." In this configuration the master glove is integrated with our previously developed non-anthropomorphic six degree-of-freedom force-reflecting hand controller (FRHC). The mechanical hand and forearm are mounted to an industrial robot (PUMA 560), replacing its standard forearm. The notion of "terminus control mode" refers to the fact that only the terminus devices (glove and robot hand) are of anthropomorphic nature, and the master and slave arms are non-anthropomorphic. The system is controlled by a high performance distributed computer controller. Control electronics and computing architecture were custom developed for this telemanipulation system, The system is currently being evaluated, focusing on tool handling and astronaut equivalent task executions. The evaluation revealed the system's potential for tool handling but it also became evident that EVA tool handling operations in space require a dexterous, human-equivalent dual arm robot,

26

The anthropomorphic telemanipulation system in terminus control configuration is shown in Figure 20. The master arm/glove and the slave arm/hand have 22 active joints each. The manipulator arm has five additional drives to control finger and wrist compliance. This Active Electromechanical Compliance (AEC) system provides the muscle equivalent dual function of position as well as stiffness control. A cable links the forearm a to an overhead gravity balance'.' suspension system, relieving the PUMA upper arm of this additional weight. The forearm has two sections, a rectangular and a cylindrical. The cylindrical section, extending beyond the elbow joint, contains the wrist actuation system. The rectangular cross section houses the finger drive actuators, all sensors and the local control and computational electronics. The wrist has three DOF with angular displacements similar to the human wrist. The wrist is linked to an AEC system that controls the wrist's stiffness. It is noted that the slave hand, wrist and forearm forma mechanically closed system, that is, the hand cannot be used without its wrist. A glove-type device is worn by the operator. Its force sensors enable hybrid position/force control and compliance control of the mechanical hand. Four fingers are instrumented, each having four d.o.f. Position feedback from the mechanical hand is providing position control for each of the 16 glove joints. The glove's feedback actuators are remotely located and linked to the glove through flex cables. A one-to-one kinematic mapping exists between master glove and slave hand joints, thus reducing the computational efforts and control complexity of the terminus subsystem. The exceptions to the direct mapping are the two thumb base joints which need kinematic transformations.

The present control electronics architecture for the master glove and the anthropomorphic hand/wrist is shown in Figure 21. It is comprised of PC based computational engines, using TMS320C40 (C40) processors and 2 custom designed intelligent controllers. The interface to the FRHC and the PUMA upper arm joints is provided by two separate Universal Motor Controllers (UMC). The UMC has been described previously in [9], The C40S communicate with each other via a single duplex communication channel. The intelligent controllers are based on the Texas

27

Instrument TMS320C30 (C30). The C30 was selected for this task because of its low cost and high performance (33 MFLOPS). The C30 is very similar to the C40 except that it lacks the 6 high speed communication ports. The two intelligent controllers are placed near the systems's sensors, one is near the master glove, the other is near the anthropomorphic hand and wrist. The function of the controllers is to provide sampling of analog signals, filtering of these signals, to provide digital calibration of strain gages, modeling the actuator voltage-velocity curve, the to generation of PWM signals, and to communicate with the PC based computational engine. All programs were written in the C language, using the SPOX Real-Time Operating System (Spectrum Microsystems) to facilitate the development of multi-purpose programs. More on this system can be found in [26].

Testing and evaluation of this system is still in progress. It became clear during the tests, however, that tool handling EVA tasks require a dual-arm fingered hand system with at least four fingers and with 7 d.o.f. compliant arms.

REMARKS

It was not possible to include all aspects of the JPL ATOP work in this Chapter. Interested readers can find useful contributions to advanced teleoperator technology in the area of kinesthetic force feedback in microgravity, hand controllers, end effectors, and stereo vision listed in references [271 through [34].

and parallel computation in telerobotics

ACKNOWLEDGEMENT

28

REFERENCES

[1]   A.K. Bejczy and J.K. Salisbury, Controlling remote manipulators through kinesthetic coupling, Computers in *Mechanical Engineering,* 1 (1), July 1983. Also, Kinesthetic Coupling Between Operator and Remote Manipulator in *Proc. ASME Int'l Computer Technology Conference Vol. 1,* San Francisco, CA, August 12-15,1980, pp. 197-211.

[2]   A.K. Bejczy, Teleoperation: The Language of the Human Hand, *Proceedings of IEEE Workshop on Robot and Human Communication,* Tokyo, Japan, September 1-3, 1992.

[3]   L.W. Knight and D. Retter, Datahand™: Design, potential performance, and improvements in the computer keyboard and mouse. In *National Human Factors Society Conference,* Denver, CO, November 1989.

[4]   L.W. Knight, Single operator environment: Experimental hand-grip controller for ATOP, Jet Propulsion Laboratory, DOE Summer Faculty Fellow (SFF) Reports, July 31,1992, and July 30,1993.

[5]   A.K. Bejczy, Z. Szakaly, and W.S. Kim, A laboratory breadboard system for dual arm teleoperation, in *Third Annual Workshop on Space Operations, Automation and Robotics,* NASA Conference Publications 3059, JSC, Houston, TX, July 25-27,1989.

[6]   A.K. Bejczy and Z. Szakaly, Performance capabilities of a JPL dual-arm advanced teleoperation system, in *Space Operations, Applications, and Research Symposium (SOAR '90) Proceedings,* Albuquerque, NM, June 26,1990.

[7]   A.K. Bejczy, Z. Szakaly, and T. Ohm, Impact of end effecter technology on telemanipulation performance, in *Third Annual Workshop on Space Operations, Automation and* Robotics, NASA Conference Publication 3059, JSC, Houston, TX, July 25-27,1989.

[8]   A.K. Bejczy and Z. Szakaly, An 8-d.o.f. dual arm system for advanced teleoperation performance experiments, in *Space Operations, Applications, and Research Symposyum (SOAR '91),* Houston, TX, July 1991. See also, Lee, S. and Bejczy, A.K., Redundant arm kinematic control based on parametrization, in *Proceedings IEEE Int'l Conference on Robotics and Automation,* Sacramento, CA, April 1991.

[9]   Bejczy, A. K., Szakal y, Z. F., Universal Computer Control System (UCCS) for Space Telerobots, *Proceedings of the IEEE Int'l Conference on Robotics and Automation,* Raleigh, NC, March 30- April 3, 1987, and see also Szakaly, Z. Fleischer, G., JPL Advanced Teleoperation Control System Critical Path Performance, JPL Memo 3470-90-332.

[10] Bejczy, A. K., Szakaly, Z., A Harmonic Motion Generator for Telerobotic Applications, *Proceedings of IEEE Int'l Conference on* Robotics *and Automation,* Sacramento, CA, April 9-11, 1991, pp. 2032-2039.

[11] A.K. Bejczy, W.S. Kim and S. Venerna, The phantom robot: Predictive display for teleoperation with time delay, in *Proceedings of IEEE International Conference on Robotics and Automation,* Cincinnati, OH, May 1990.

[12] A.K. Bejczy and W.S. Kim, Predictive displays and shared compliance control for time delayed telemanipulation, in Proceedings of *IEEE Int'l Workshop on Intelligent Robots and Systems (IROS '90),* Tsuchiura, Japan, July 1990.

[13] W.S. Kim and A.K. Bejczy, Graphics displays for operator aid in telemanipulation, in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics,* Charlottesville, VA, October 1991.

[14] W.S. Kim, Graphical operator interface for space telerobotics, in Proceedings of *IEEE International Conference on Robotics and Automation,* Atlanta, GA, May 1993.

[15] P. Fiorini, A.K. Bejczy, and P. Schenker, Integrated interface for advanced teleoperation, *IEEE Control Systems Magazine, 13(5),* October 1993.

[16] W.S. Kim and A.K. Bejczy, Demonstration of a high-fidelity predictive/preview display technique for telerobotics servicing in space, IEEE *Transaction on Robotics and Automation,* October 1993. Special Issue on Space Telerobotics. See also, W.S. Kim, P.S. Schenker, A.K. Bejczy, S. Leake, and S. Ollendorf, "An Advanced Operator Interface Design with Preview/Predictive Displays for Ground-Controlled Space Telerobotic Servicing," *SPIE Conference 2057: Telemanipulator Technology and Space Telerobotics,* Boston, MA, September 1993.

[17] W.S. Kim, Virtual reality calibration for telerobotic servicing, in Proceedings *of IEEE International Conference on Robotics and Automation,* San Diego, CA, May 1994.

[18] P. Lee, B. Hannaford, and L. Wood, Telerobotic configuration editor, in *Proceedings of IEEE International Conference on Systems, Man and Cybernetics, Los Angeles,* CA, 1990.

[19] B. Hannaford et al, Performance evaluation of a six-axis generalized force-reflecting teleoperator, *JPL Publication 89-18,* Jet Propulsion Laboratory, June 15, 1989.

[20] B. Hannaford et al, Performance evaluation of a six-axis force-reflecting teleoperation, *IEEE Transaction on Systems, Man and Cybernetics, 21(3),* May/June 1991.

[21] H. Das et al, Performance experiments with alternative advanced teleoperator control modes for a simulated solar max satellite repair, in *Proceedings of Space Operations, Automation and Robotics Symposium (SOAR 91), JSC,* Houston, TX, July 9-11, 1991.

[22] H. Das et al, Performance with alternative control modes in teleoperation, *PRESENCE: Teleoperators and Virtual Environments, 1(2),* Spring 1993. MIT Press Publication.

[23] P. Fiorini et al, Nueral networks for segmentation of teleoperation tasks, *PRESENCE: Teleoperators and Virtual Environments, 2(l),* Winter 1993. MIT Press Publication.

[24] B. Hannaford and P. Lee, Hidden markov model analysis of force-torque information in telemanipulation, *International journal of Robotics Research, 10(5),* October 1991.

[25] Jau, B,M., "Man-Equivalent Teleopresence Through Four Fingered Human-Like Hand System", Proceedings of *IEEE Int'l Conference on Robotics and Automation,* Nice, France, May 12-14, 1992, pp 843-848.

[26] B.M. Jau, M. Anthony Lewis and A.K. Bejczy, Anthropomírphlic Telemanipulation System in Terminus Control Mode, *Proceedings of ROMANSY '94,* Gdansk, Poland, September 12-16,1994, Springer Verdag.

[27J Kevin Corker, Investigation of Neuromotor *Control* and Sensory Sampling in Bilateral Teleoperation, Ph.D. Thesis, University of California, Los Angeles, CA, 1984.

[28] A.K. Bejczy and K. Corker, Manual Control Communication in Space Teleoperation, *Proceedings of Fifth CISM-IFTOMM Symposium on Theory and Practice of Robots and Manipulators;* publication R. Kogan Page Ltd., London and Hermes Publishing, Paris , 1985.

[29] T.L. Brooks and A.K. Bejczy, Hand Controllers for Teleoperation: A State-of-the-Art Survey and Evaluation, JPL Publication 85-11,1985.

[30] A.H. Mishkin and B.M. Jau, Space-Based Multifunctional End Effecter Systems, JPL Publication 88-16, 1988.

[31] D.B. Diner and Marika von Sydow, Stereo Depth Distortions in Teleoperation, JPL Publication 87-1, Rev. 1,1988.

[32] D.B. Diner and D.H. Fender, Human Engineering in Stereoscopic Viewing Series, JPL D-8186,1991.

[33] D.B. Diner, A new Definition of Orthostereopsis for 3-D Television, *Proceedings of IEEE Int 'L Conference on Systems, Man, and Cybernetics,* Charlottesville, VA, October 1991.

[34] A. Fijany and A. Bejczy, Eds., Parallel Computation Systems for Robotics; Algorithms and Architectures, Book by World Scientific Publication, 1992.
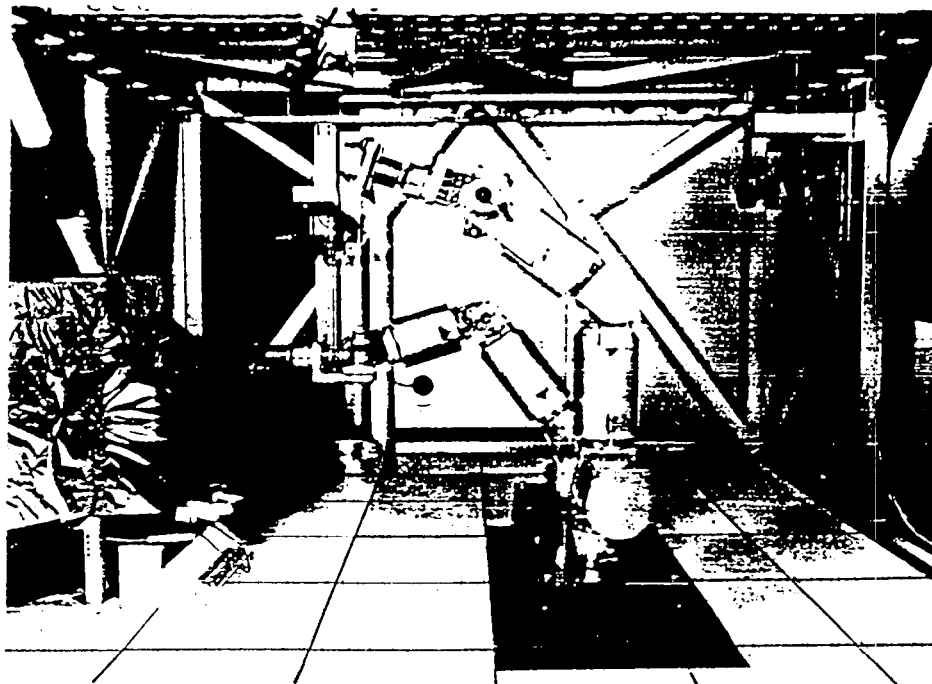
JPL - 19467

Figure 1: The JPL ATOP Dual Arm Workcell with Gantry TV Frame

JPL- 19902.1

Figure 2: The JPL ATOP Control Station

On-Line Robotic Control
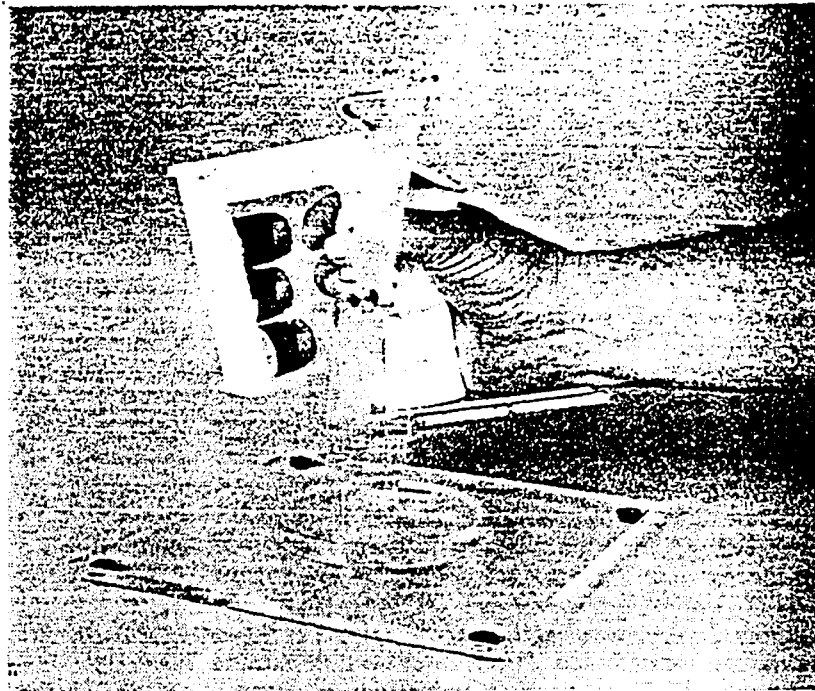
JPL-19465.A

Off-Line TCE Control
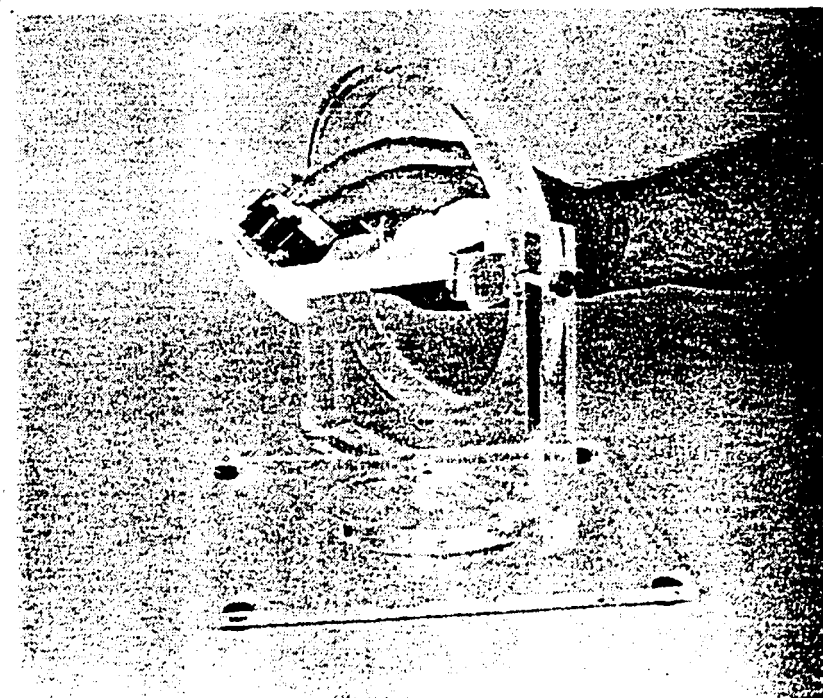
JPL-19467 A

Figure 5. IDATAHAND Switch Modules Integrated with FRHC Hand-Grip

30 cm (12 in.)
CuBE

WORKING VOLUME

1 meter

2-AXIS
HANDGRIP

10 LER/COUNTERBALANCE

POWER JINIT

GIMBBIAED D
SUPPORT
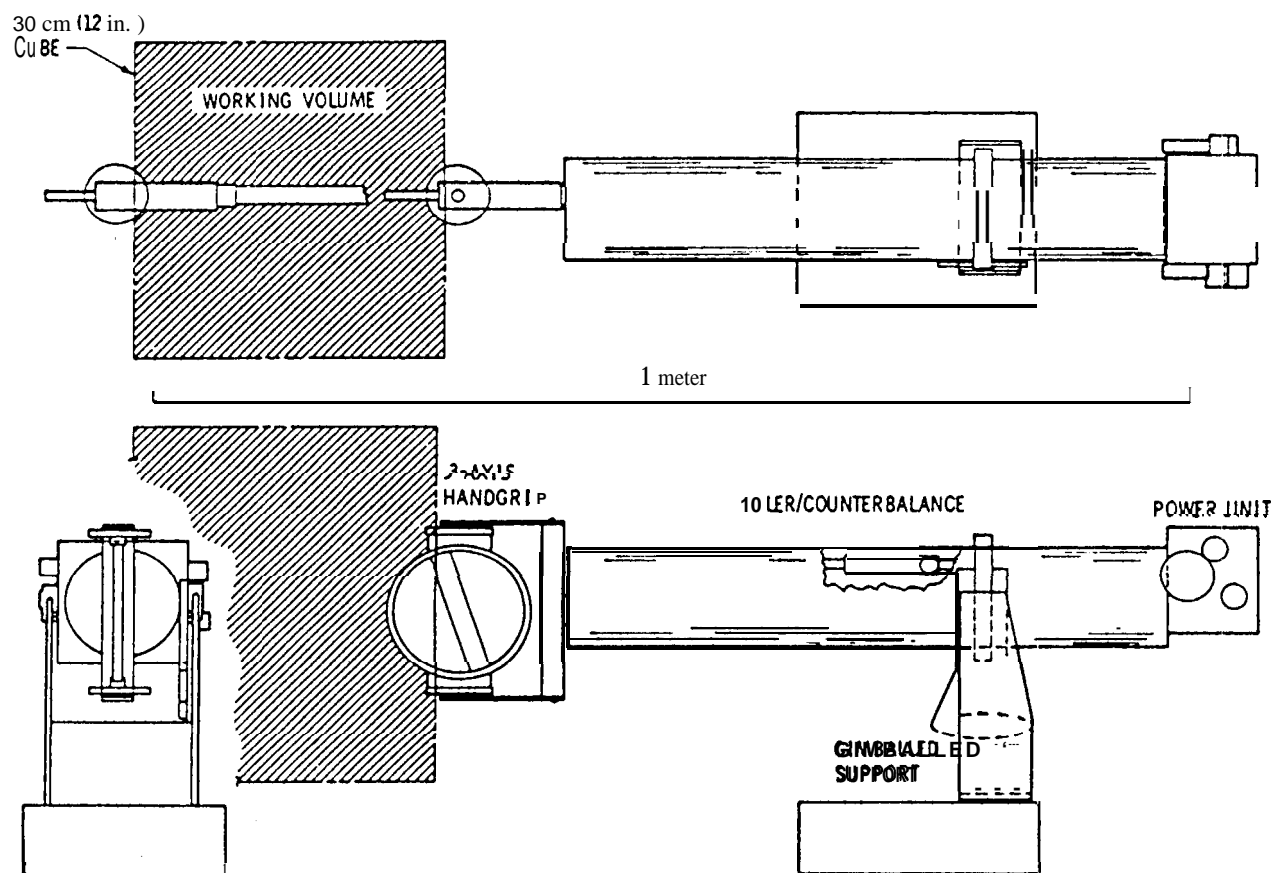
Fig. Six-axis force-reflecting hand controller overall schematic.

3

RANGE OF MOTION

$\beta_4, \beta_5, \beta_6$: ±180°

R_3: 0-13"

β_1, β_2: ±20°
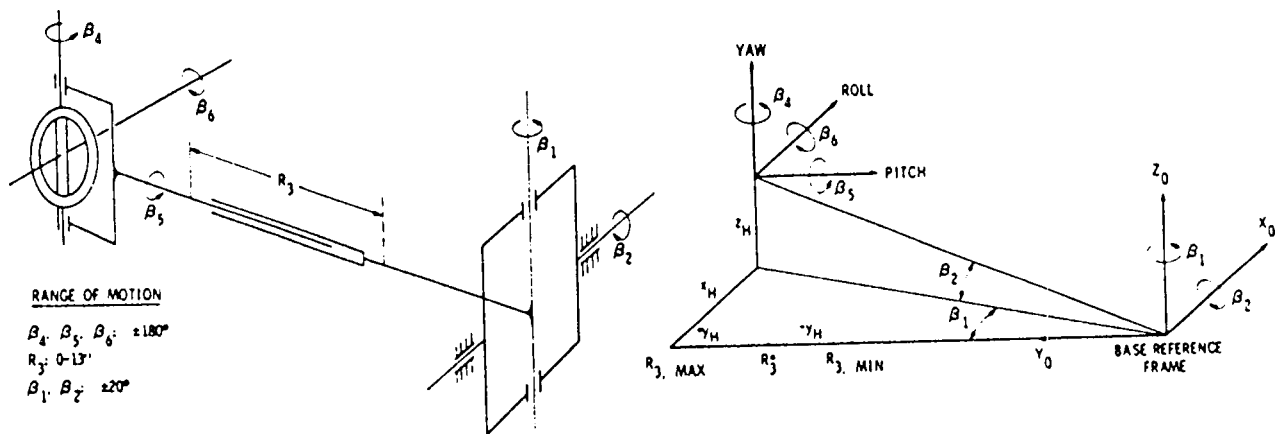
Fig. 6. Hand controller kinematics and command axes.
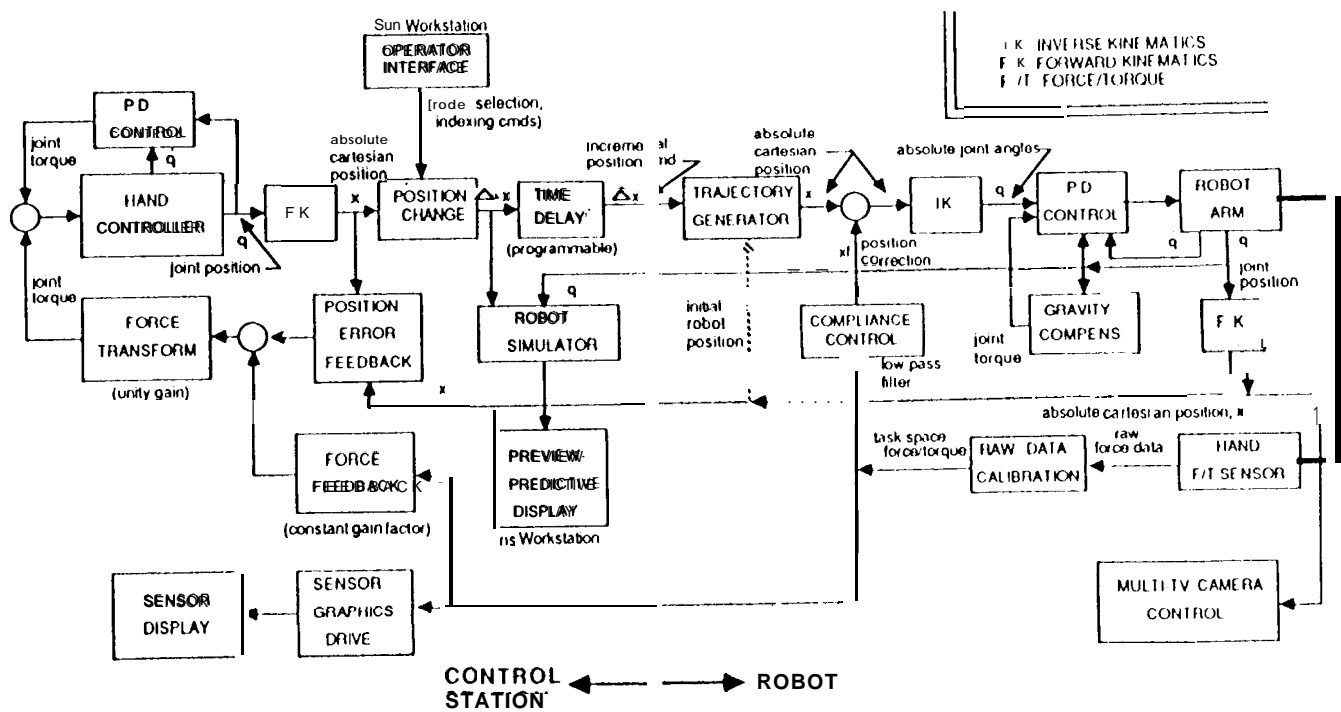
Figure 7. System Flow Diagram
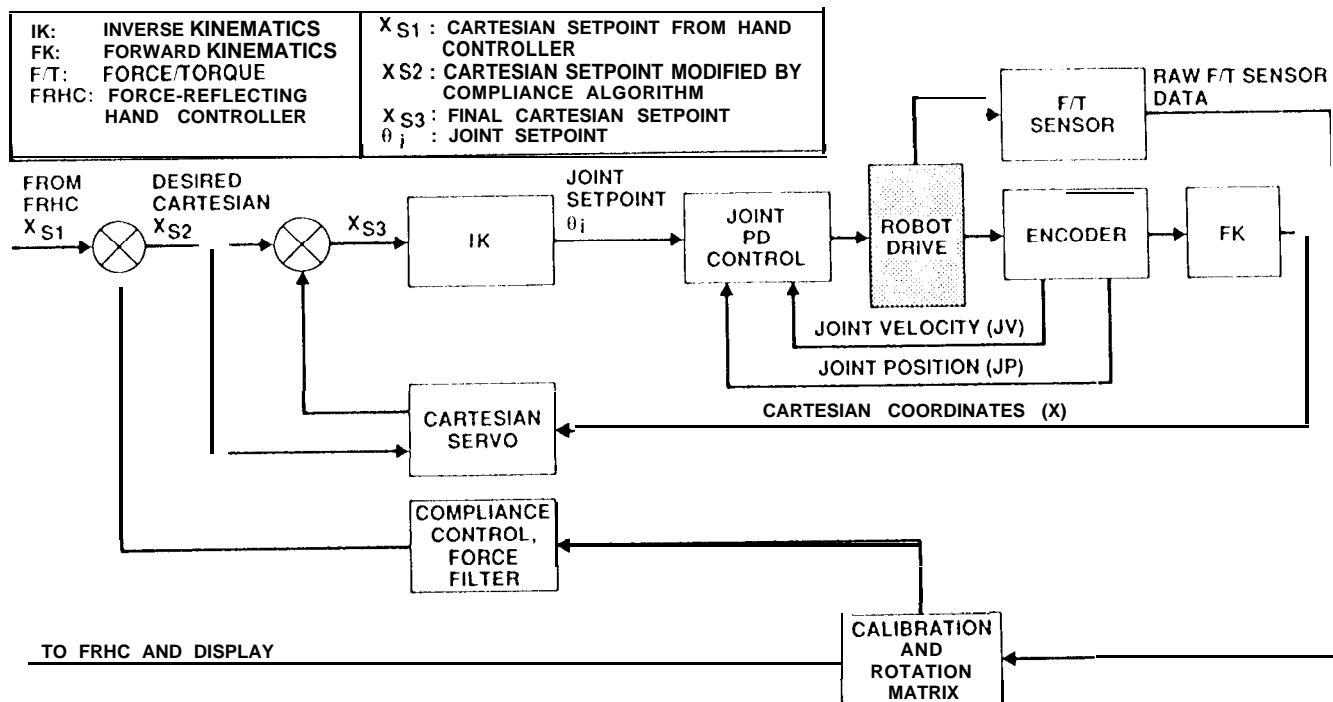
Figure 8. Control Schemes: Joint Servo, Cartesian Servo, Compliance Control
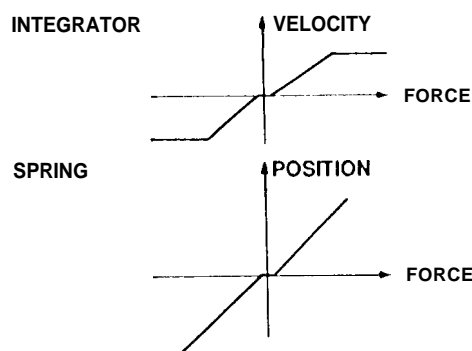
Figure 9. Compliance Components
and Interpretations
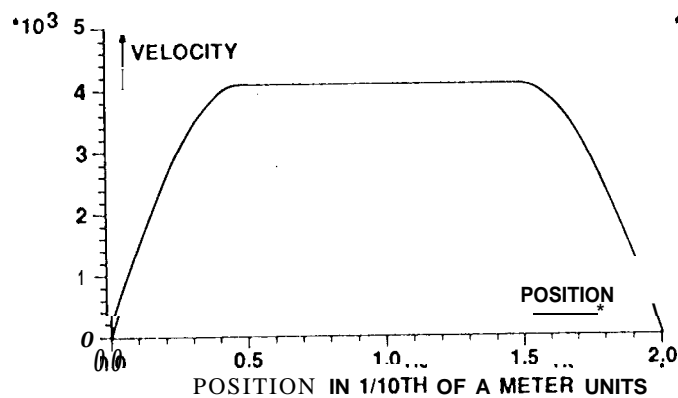
POSITION IN 1/10TH OF A METER UNITS
Figure 10. Harmonic Motion Generator
Velocity-Position Function
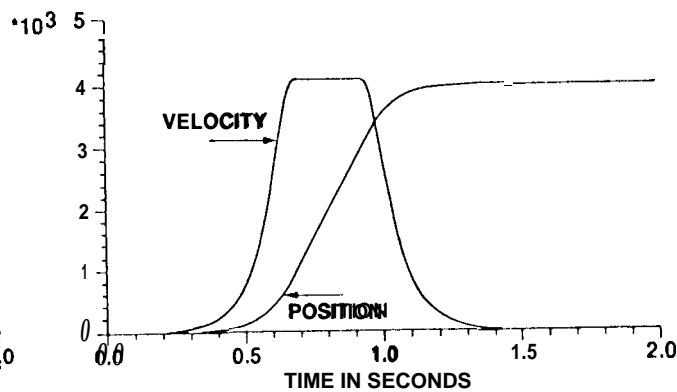
TIME IN SECONDS
Figure 11. Harmonic Motion Generator Position and
Velocity lime Functions

Figure 12. Vertical (Z) Straight Line Trajectory and AX Error.

Figure 13. Horizontal   Straight Line Trajectory from Harmonic Motion Genrator and ΔX & ΔZ Errors

P—41862. A

Figure 14. Graphics User Interface for Calibrating Virtual (Graphics) Images to TV Images
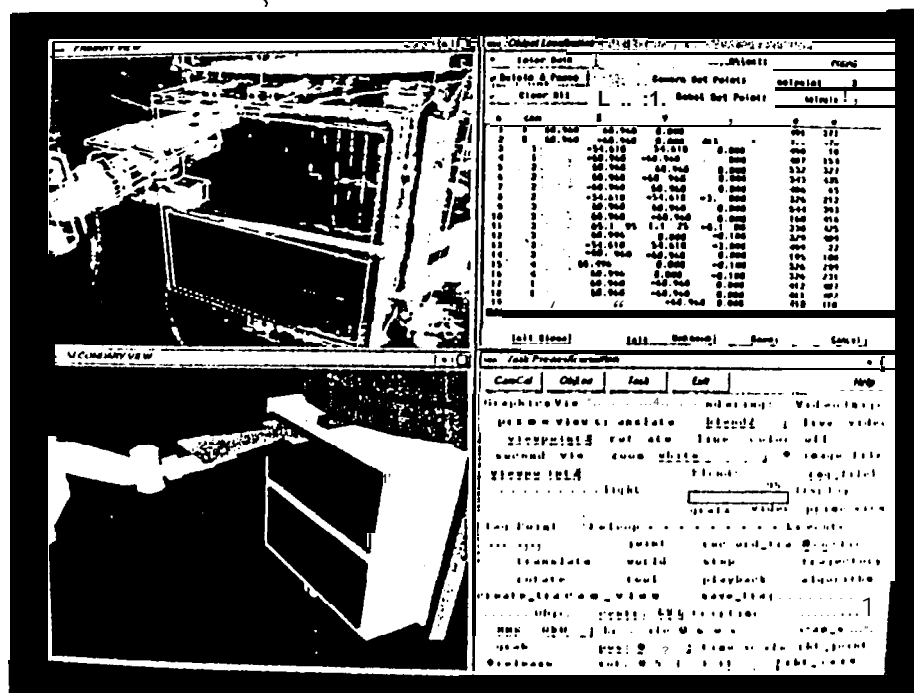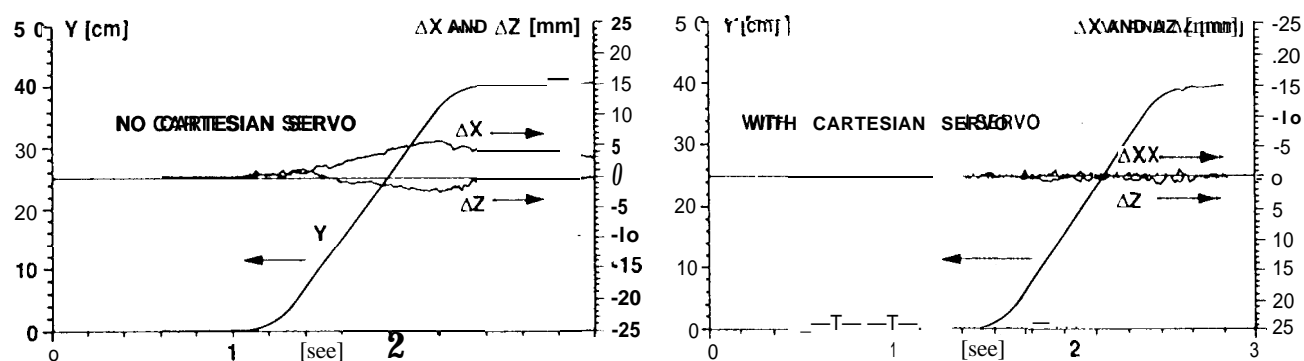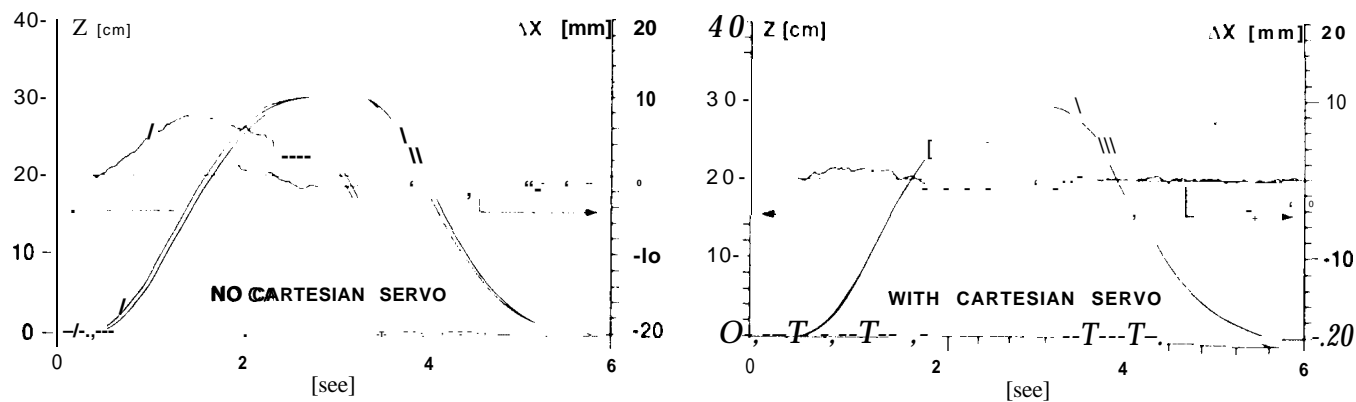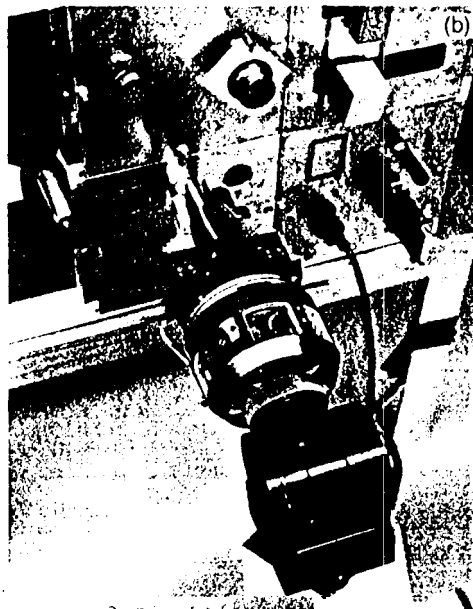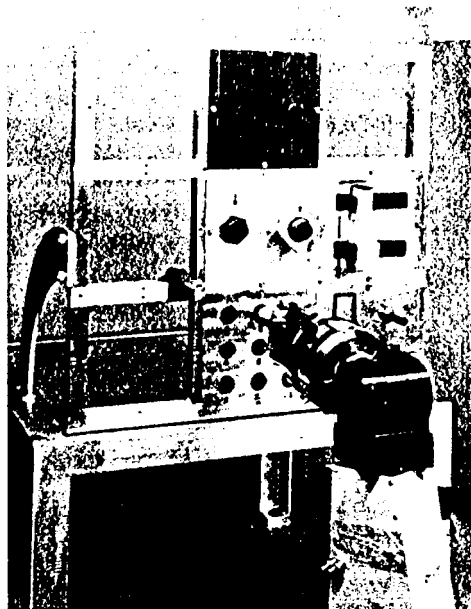
Figure 17. Two views of the task board and the robot hand performing the peg-in-hole task. The modular task board design has nine 7" x 7" openings for task modules. Shown are the four task modules used in the experiments: (a) Bayonet Connector and Peg-in-hole tasks (upper and lower left modules); (b) Velcro and Electrical Connectors tasks (upper and lower right modules). Task modules could be removed and mounted on a separate force-torque sensor for measurement of force and torques in hand operation.
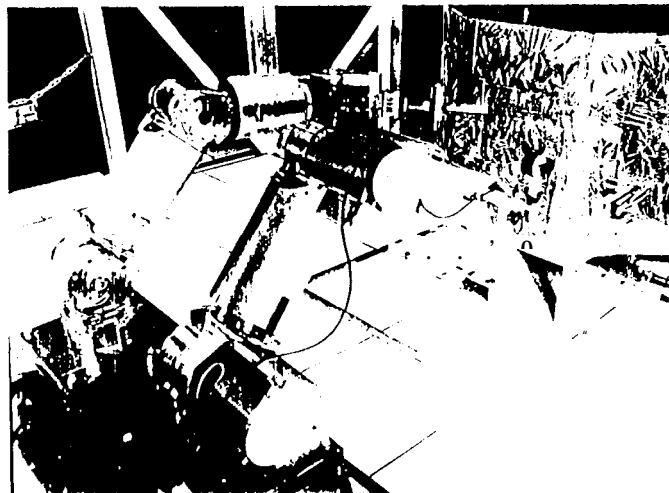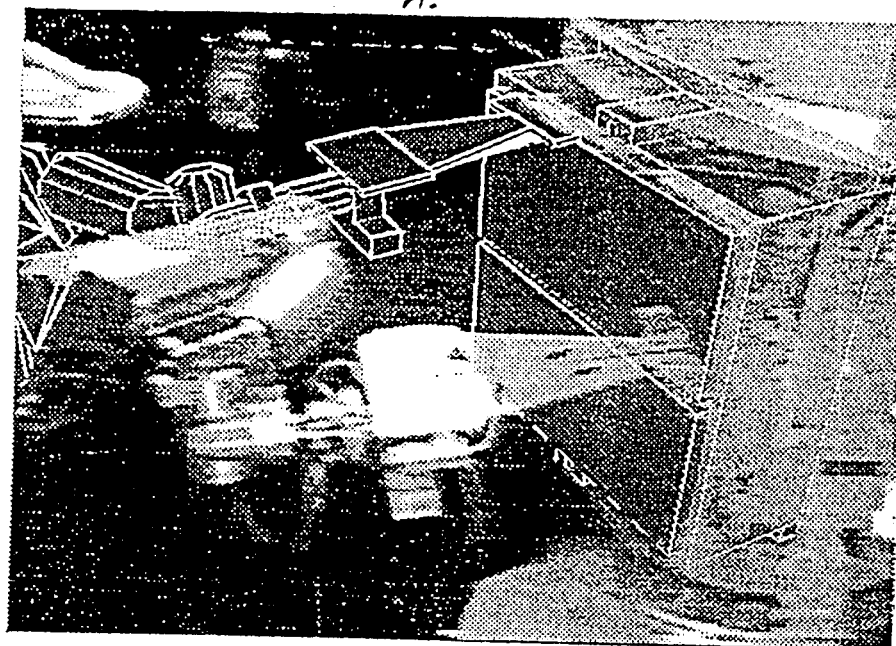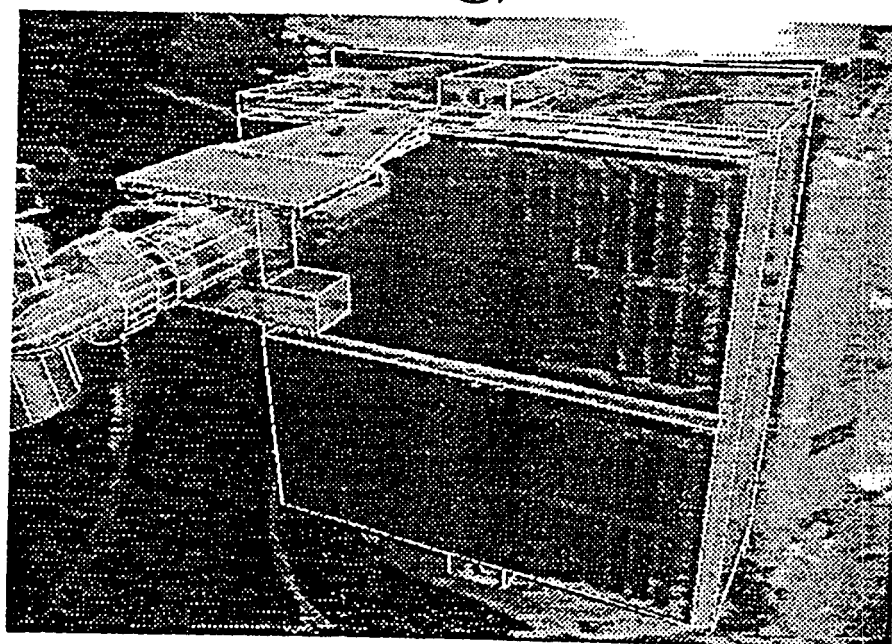
Figure 18. An SMSR repair subtask simulation; reinstating the satellite's thermal blanket.
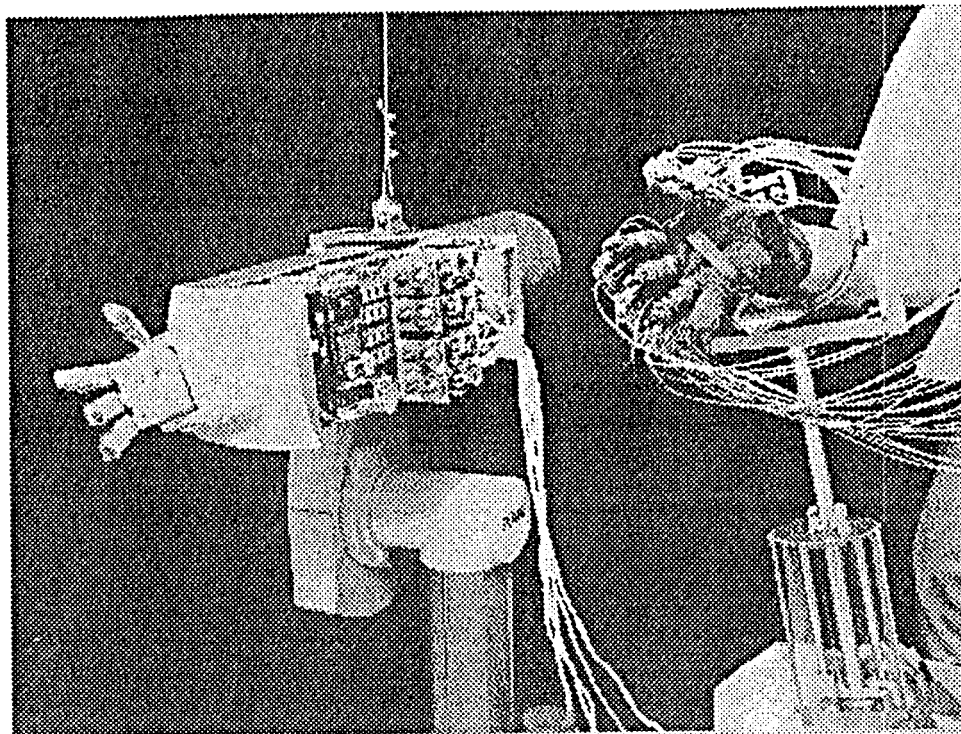
A.

WSK

B.

WSK

Figure 19. A. Predictive/Preview Display of End Point Motion. B. Status of Predicted End Point after Motion Execution, from a Different Camera View, for the Same Motion Shown Above.
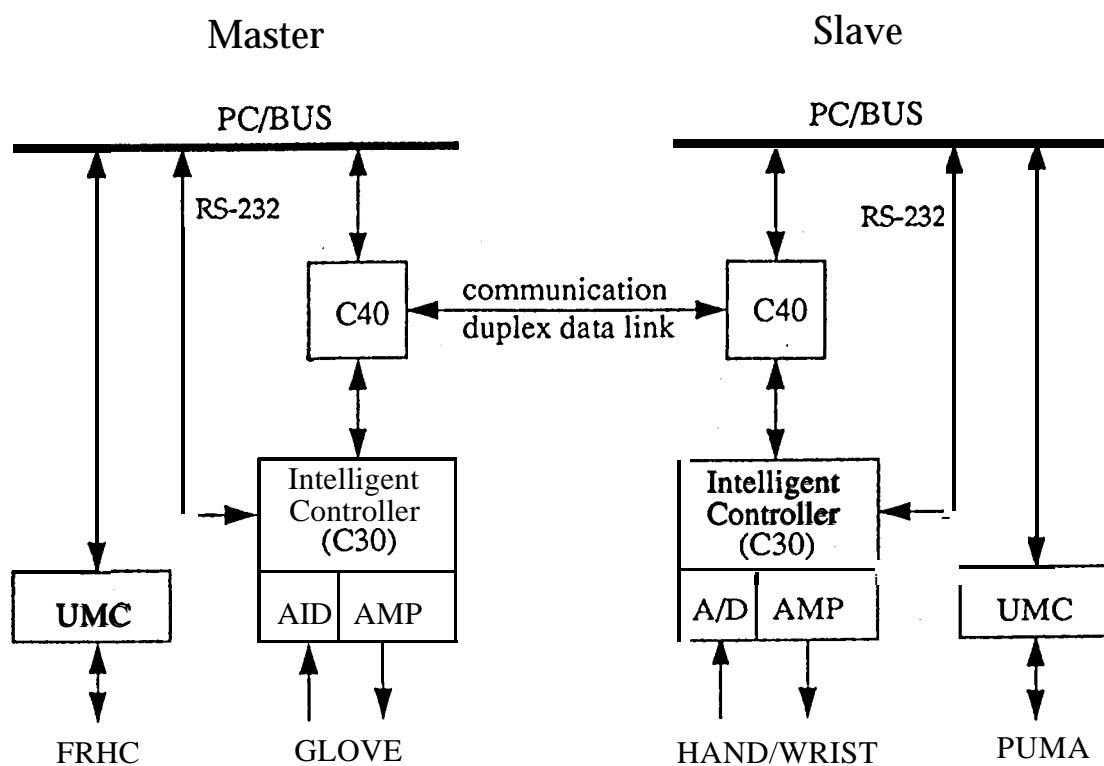
Fig. 20. The Master Glove Controller and the Anthropomorphic Hand

Fig. 21. Control Architecture Overview